(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 3 913 850 A1

(12) EUROPEAN PATENT APPLICATION
published in accordance with Art. 153(4) EPC

(43) Date of publication:
24.11.2021 Bulletin 2021/47

(21) Application number: **20814904.7**

(22) Date of filing: **19.05.2020**

(51) Int Cl.:
H04L 9/00 *(2006.01)*

(86) International application number:
PCT/CN2020/091002

(87) International publication number:
WO 2020/238694 (03.12.2020 Gazette 2020/49)

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME
Designated Validation States:
KH MA MD TN

(30) Priority: **27.05.2019 CN 201910445155**

(71) Applicant: **TENCENT TECHNOLOGY (SHENZHEN) COMPANY LIMITED**
Shenzhen, GUANGDONG 518057, (CN)

(72) Inventor: **KONG, Qinglei**
Shenzhen, Guangdong 518057 (CN)

(74) Representative: **EP&C**
P.O. Box 3241
2280 GE Rijswijk (NL)

(54) KEY MANAGEMENT METHOD AND RELATED DEVICE

(57) Disclosed are a key management method and a related device. The method comprises: a client selects a random number and a key according to an operation instruction inputted by a user; generate a first encrypted ciphertext of the key according to the random number, the key, a first public key, and a second public key, wherein the first public key is determined according to a point on an elliptic curve and a private key of a hardware security module, and the second public key is determined according to the point on the elliptic curve and a private key of the client; generate a symmetric key sequence according to the key and a preselected Hash function, and according to the symmetric key sequence, encrypt cloud storage data to obtain a data ciphertext; and transmit the first encrypted ciphertext and the data ciphertext to a cloud server.

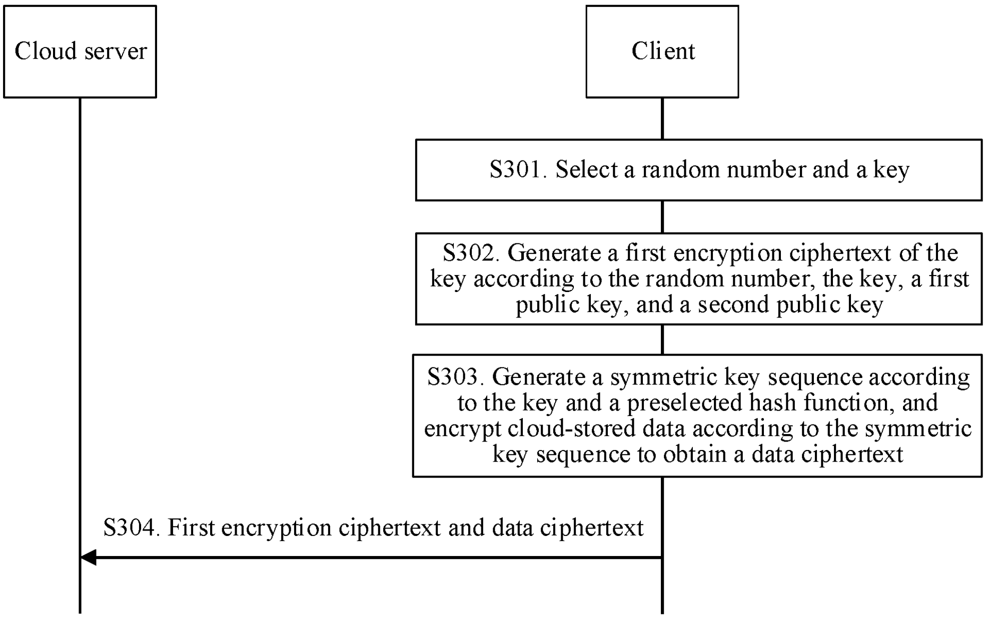
Cloud server
Client
S301. Select a random number and a key
S302. Generate a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key
S303. Generate a symmetric key sequence according to the key and a preselected hash function, and encrypt cloud-stored data according to the symmetric key sequence to obtain a data ciphertext
S304. First encryption ciphertext and data ciphertext

FIG. 3

Printed by Jouve, 75001 PARIS (FR)

Description

RELATED APPLICATION

[0001] This application claims priority to Chinese Patent Application No. 201910445155.9, entitled "KEY MANAGEMENT METHOD AND RELATED DEVICE" filed on May 27, 2019, which is incorporated herein by reference in its entirety.

FIELD OF THE TECHNOLOGY

[0002] This disclosure relates to the field of security technologies, and in particular, to a key management method and a related device.

BACKGROUND OF THE DISCLOSURE

[0003] Key management service (KMS) is a cloud key escrow service that aims at creating and controlling encryption keys required for encrypted data and that is integrated with other cloud servers, to enable the cloud servers to manage the encrypted data by using the encryption keys.

SUMMARY

[0004] Embodiments of this disclosure provide a key management method and a related device, to improve efficiency in generation and storage of a cloud-stored key, and ensure security of cloud-stored data.

[0005] According to a first aspect, an embodiment of this disclosure provides a key management method, including:

selecting, by a client, a random number and a key according to an operation instruction inputted by a user;

generating, by the client, a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client;

generating, by the client, a symmetric key sequence according to the key and a preselected hash function, and encrypting cloud-stored data according to the symmetric key sequence to obtain a data ciphertext; and

transmitting, by the client, the first encryption ciphertext and the data ciphertext to a cloud server.

[0006] According to a second aspect, an embodiment of this disclosure provides another key management method, including:
receiving, by a cloud server, a first encryption ciphertext and a data ciphertext transmitted by a client, the first encryption ciphertext being generated according to a random number, a key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, the second public key being determined according to the point on the elliptic curve and a private key of the client, the data ciphertext being obtained by encrypting cloud-stored data according to a symmetric key sequence, and the symmetric key sequence being generated according to the key and a preselected hash function; and storing, by the cloud server, the first encryption ciphertext and the data ciphertext.

[0007] According to a third aspect, an embodiment of this disclosure provides a client, including:

a selection module, configured to select a random number and a key according to an operation instruction inputted by a user;

a processing module, configured to generate a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client,

the processing module being further configured to generate a symmetric key sequence according to the key and a preselected hash function, and encrypt cloud-stored data according to the symmetric key sequence to obtain a data ciphertext; and a transmitting module, configured to transmit the first encryption ciphertext and the data ciphertext to a cloud server.

[0008] According to a fourth aspect, an embodiment of this disclosure provides a cloud server, including:
a receiving module, configured to receive a first encryption ciphertext and a data ciphertext transmitted by a client, the first encryption ciphertext being generated according to a random number, a key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, the second public key being determined according to the point on the elliptic curve and a private key of the client, the data ciphertext being obtained by encrypting cloud-stored data according to a symmetric key sequence, and the symmetric key sequence being generated according to the key and a preselected hash function.

[0009] According to a fifth aspect, an embodiment of this disclosure provides another client, including a processor, a memory, and a communication bus, the communication bus being configured to implement connection and communication between the processor and the memory, and the processor executing a program stored in the memory, to implement the steps in the key management method according to the first aspect.

[0010] In a possible design, the client provided in this disclosure may include a corresponding module configured to perform an action of the client in the foregoing method design. The module may be software and/or hardware.

[0011] According to a sixth aspect, an embodiment of this disclosure provides another cloud server, including a processor, a memory, and a communication bus, the communication bus being configured to implement connection and communication between the processor and the memory, and the processor executing a program stored in the memory, to implement the steps in the key management method according to the second aspect.

[0012] In a possible design, the cloud server provided in this disclosure may include a corresponding module configured to perform an action of the cloud server in the foregoing method design. The module may be software and/or hardware.

[0013] According to a seventh aspect, an embodiment of this disclosure provides a computer-readable storage medium, the computer-readable storage medium storing instructions, the instructions, when run on a computer, causing the computer to perform the method according to the foregoing aspects.

[0014] According to an eighth aspect, an embodiment of this disclosure provides a computer program product including instructions, when run on a computer, the computer program product causing the computer to perform the method according to the foregoing aspects.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] To describe technical solutions in embodiments of this disclosure more clearly, the following briefly introduces the accompanying drawings required for describing the embodiments. Apparently, the accompanying drawings in the following description show some embodiments of this disclosure, and a person of ordinary skill in the art may still derive other accompanying drawings from these accompanying drawings without creative efforts.

FIG. 1 is a schematic diagram of a key management method according to an existing technical solution.

FIG. 2 is a schematic architectural diagram of a key management system according to an embodiment of this disclosure.

FIG. 3 is a schematic flowchart of a key management method according to an embodiment of this disclosure.

FIG. 4 is a schematic diagram of a storage format of an encryption ciphertext and a data ciphertext according to an embodiment of this disclosure.

FIG. 5 is a schematic flowchart of another key management method according to an embodiment of this disclosure.

FIG. 6 is a schematic diagram of storage load comparison according to an embodiment of this disclosure.

FIG. 7 is a schematic structural diagram of a client according to an embodiment of this disclosure.

FIG. 8 is a schematic structural diagram of a cloud server according to an embodiment of this disclosure.

FIG. 9 is a schematic structural diagram of another client according to an embodiment of this disclosure.

FIG. 10 is a schematic structural diagram of another cloud server according to an embodiment of this disclosure.

DESCRIPTION OF EMBODIMENTS

[0016] The following clearly and completely describes the technical solutions in the embodiments of this disclosure with reference to the accompanying drawings in the embodiments of this disclosure. Apparently, the described embodiments are some rather than all of the embodiments of this disclosure. All other embodiments obtained by a person of ordinary skill in the art based on the embodiments of this disclosure without making creative efforts shall fall within the protection scope of this disclosure.

[0017] For ease of understanding, the terms are explained below.

1. Key management service (KMS): a secure, easy-to-use key generation and management service. A final objective of a KMS is to protect the security of statically-stored data of a user, that is, the confidentiality, the integrity, and the availability of the data of the user. The KMS is deployed on a cloud server to allow a user to securely and conveniently use and manage a key for data protection and data encryption and decryption, and focus on scenarios of static data encryption and decryption of the cloud server.

2. Hardware security module (HSM): a computer hardware device configured to guarantee and manage a digital key used by a strong authentication system and also provide a cryptography-related operation, and is connected directly to a computer and a network server through an expansion card and an external device. Because the HSM can provide, for an application program, services of protecting an encryption key, and configuring encryption, decryption, identity authentication, and a digital signature, the HSM of the cloud server can provide protection for storage of a root key of data. In addition, the HSM provides two tamper-proof functions, that is, tamper evidence and tamper resistance functions.

3. Hash function: also referred to as a hashing function. The hash function converts an input of an arbitrary length into an output of a fixed length by using a hashing algorithm, and includes the following basic properties:

(1) Pre-image: This property indicates one-wayness of an operation direction of the hash function. Given a hash value h, corresponding information m(h=hash(m)) cannot be found, that is, only an output can be derived from an input, but an input cannot be calculated from an output.

(2) Second pre-image: This property means that an input of which an output result is caused to be equal to a known output result cannot be found. Given an input m1, another input m2 that causes hash values to be hash(m1)=hash(m2) cannot be found.

(3) Collision resistance: This property means that two different inputs m1 and m2 of which output results hash(m1) and hash(m2) are caused to be the same, hash (m1)=hash(m2), cannot be found.

4. Elliptic curve cryptography (ECC): a mathematical public-key encryption algorithm based on elliptic curves. The mathematical basis of the ECC is the computational difficulty in constituting elliptic discrete logarithms on an abelian additive group by using rational points on an elliptic curve. A main advantage of the ECC lies in using smaller keys compared to another method (for example, the RSA encryption algorithm) in some cases to provide equivalent or higher-level security.

5. Symmetric-key encryption (SKE): an encryption algorithm in which a same key needs to be used for encryption and decryption. Due to a high speed thereof, the SKE is generally used when a message sender needs to encrypt a large amount of data. Advanced Encryption Standard (AES) is a block encryption standard using a substitution-permutation network, and serves as one of the most popular algorithms in SKE.

[0018] Use of the KMS mainly includes two scenarios: (1) using a master key stored on the KMS to encrypt/decrypt data by calling an application programming interface (API) of the KMS; and (2) using a data key in a cloud server to encrypt/decrypt data, and using a master key in the KMS to protect the data key. In the KMS, the master key is protected by an HSM, and the master key can only be used in the HSM to ensure availability, security and durability of the key. In addition, the KMS supports a customer to create a master key and import the master key into the HSM.

[0019] FIG. 1 is a schematic diagram of a key management method according to an existing technical solution. A data encryption key (DEK) is protected by an HSM backing key (HBK), the HBK is protected by a domain key (DK), and the DK is protected by an HSM domain key encryption key (DKEK) that is permanently stored, and the DKEK is stored on an HSM. Table 1 is a KMS key management structure. The DKEK is an outermost encryption key, the DK is a second outermost encryption key, the HBK is an inner encryption key, and the DEK is an innermost encryption key. The DK, the HBK and the DEK are all generated by the HSM.

Table 1

	Generated by	Upper encryption key	Description
DK	HSM	DKEK	Permanently stored on HSM
HBK	HSM	DK	Stored in cloud storage
DEK	HSM	HBK	One-time pad

[0020] The existing KMS key management solutions have the following technical problems: (1) In view of the sensitivity of current customers to data privacy protection, a data key and encrypted data are both managed by a cloud server. Even if a cloud service provider strictly controls and supervises equipment maintenance and management personnel, a scenario in which an insider leaks and steals data still often occurs. (2) Because an encrypted data key and encrypted data are both stored on a cloud storage server, costs of key management and encryption key storage of the cloud server are increased. (3) Because a layer-by-layer wrapping structure between keys is used, costs of key decryption and key storage of the HSM are also increased. Because of limited computing and storage capabilities of the HSM, costs of equipment investment of the cloud server are increased.

[0021] FIG. 2 is a schematic architectural diagram of a key management system according to an embodiment of this disclosure. The key management system in this embodiment of this disclosure includes a cloud server 1, an HSM 2, and a client 3. The cloud server 1 is an important component of a cloud computing service, and is a service platform providing comprehensive service capabilities to a variety of Internet users. The service platform integrates three core elements of an Internet application in the conventional sense: computing, storage, and the network, and provides public Internet infrastructure services for users. Each cluster node in the cloud server is deployed in a backbone data center of the Internet, and can independently provide Internet infrastructure services such as computing, storage, online backup, escrow, and bandwidth. The HSM 2 is an HSM of the cloud server. The HSM 2 is a computer hardware device that may be configured to guarantee and manage a digital key used by a strong authentication system and also provide a cryptography-related operation, and is connected directly to a computer and a network server through an expansion card and an external device. Because the HSM 2 can provide, for an application program, services of protecting an encryption key, configuring encryption, decryption, identity authentication, and a digital signature, the HSM 2 can provide protection for storage of a root key of data. In addition, the HSM 2 provides two tamper-proof functions, that is, tamper evidence and tamper resistance functions. The client 3 may be a smartphone, a portable computer, a handheld communication device, a handheld computing device, a satellite radio device, a global positioning system, a personal digital assistant (PDA), and/or any other appropriate device used for communication on a wireless communication system and the like. The cloud server 1, the HSM 2, and the client 3 may establish network connections to each other and communicate with each other through the established network connections.

[0022] This embodiment of this disclosure is applicable to the following scenarios: privacy protection of cloud-stored data and generation of a cloud-stored key. In this embodiment of this disclosure, the DEK is protected by using a public key of the HSM and a public key of the client, so that cloud-stored data of the client can be decrypted only when the HSM and the client perform decryption jointly.

[0023] FIG. 3 is a schematic flowchart of a key management method according to an embodiment of this disclosure. The procedure includes key generation and data encryption processes. This embodiment of this disclosure includes at least the following steps:

[0024] S301. A client selects a random number and a key according to an operation instruction inputted by a user. The random number and the key are both less than n, n being an order of a point on an elliptic curve.

[0025] In a specific implementation, for example, a user may access, through a client, a web page for creating a key on a cloud server. The client may receive an operation instruction inputted by the user through the web page for creating a key (for example, the user clicks or taps a button for generating a key on the web page), and execute code on the web page, to select any integer from a range from 1 to n-1 as a random number, and select an integer from a range from 1 to n-1 as a key, n being an order of a point on an elliptic curve. For example, for an elliptic curve $E : y^2 = x^3 + ax + b,$ if for a point P on the elliptic curve, a minimum positive integer n exists and makes n times P=O∞, n is referred to as an order of P, and if n does not exist, P is of infinite order. O∞ is an infinity point on the elliptic curve.

[0026] Exemplarily, before the client selects a random number and a key, the HSM can perform initialization on the key management system. For example, when a user creates an account of the user on the cloud server, the HSM performs the initialization on the key management system. Specifically, the HSM selects one prime number q from a plurality of prime numbers, and then, selects two non-negative integers α and b less than q from a plurality of integers, to make that $4a^3 + 27b^2 = 0 \bmod q$ is not established. Therefore, all points (x, y) that satisfy the formula $E : y^2 = x^3 + ax$

+ b and the infinite point O∞ form an elliptic curve, where x and y are integers ranging from 0 to p-1. Then, the HSM performs discretization on the elliptic curve, for example, $y^2 = x^3 + ax + b \bmod q$, to obtain all solutions (x, $y) \in Z_q$ and one infinite point O∞. Finally, the HSM selects one point P from all the solutions as any point on the elliptic curve, and an order of the point P is n. In addition, the HSM may select one hash function from a plurality of hash functions, to make the hash function satisfy a condition $H : \{0,1\}^* — > \{0,1\}^{1024}$. The hash function may map any character string to a character string of a fixed length.

[0027] Then, the HSM may select an integer s_1 from [1, n-1] as a private key of the HSM, and then calculate a first public key of the HSM according to the private key of the HSM and the point P on the elliptic curve, where the first public key is $S_1 = s_1 gP$, and may be represented as (S_1,P).

[0028] During initialization of the key management system, the client may receive the point P on the elliptic curve and n (an order of P) sent by the HSM, then select, according to a selection instruction inputted by the user (for example, the selection instruction is inputted through a web page provided by the cloud server or the HSM), select one integer s_2 from [1, n-1] as a private key of the client, and then calculate a second public key of the client according to the private key of the client and the point P on the elliptic curve, where the second public key $S_2 = s_2 gP$, and may be represented as (S_2, P). The user may download the private key of the client through the client, and store the private key in a memory of the client. The public key and the private key of the client are also referred to as a public key and a private key of the user.

[0029] S302. The client generates a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client;

[0030] In a specific implementation, the client may generate, by using the ECC, an encryption ciphertext Ek (that is, the first encryption ciphertext) of a key K according to both the first public key (S_1,P) of the HSM and the second public key (S_2,P) of the client, where Ek , for example, is:

$$Ek = (r \cdot P, (S_1 + S_2) \cdot r + k), \qquad \text{(Formula 1)}$$

where r is the random number, k is the key, S_1 is the first public key, and S_2 is the second public key. It can be learned from formula 1 that the encryption ciphertext Ek is determined jointly by the first public key of the HSM and the second public key of the client. $r \cdot P$ represents calculating a mapped point on the elliptic curve according to the random number r and the point P on the elliptic curve. The private key of the HSM is an integer selected by the HSM from a range from 1 to n-1, and the private key of the client is an integer selected by the client from a range from 1 to n-1, n being an order of the point on the elliptic curve.

[0031] S303. The client generates a symmetric key sequence according to the key and a preselected hash function, and encrypts cloud-stored data according to the symmetric key sequence to obtain a data ciphertext.

[0032] In a specific implementation, the client may call an encryption API, to generate, according to the key and a preselected hash function, a symmetric key sequence $(H(1\|k), H(2\|k),..., H(m \| k))$ used for encrypting cloud-stored data, where k is the key, and H is the hash function, and encrypt, by using the ECC, cloud-stored data $(m1,m2,...mk)$ according to the symmetric key sequence $(H(1\|k), H(2\|k),..., H(m\|k))$ to obtain a data ciphertext (Enc (m1), ..., Enc (mk)).

[0033] S304. The client transmits the first encryption ciphertext and the data ciphertext to a cloud server. The cloud server stores the first encryption ciphertext and the data ciphertext. For example, FIG. 4 is a schematic diagram of a storage format of a first encryption ciphertext and a data ciphertext according to an embodiment of this disclosure. The former portion is the first encryption ciphertext, and the latter portion is the data ciphertext. The data ciphertext can be decrypted by using the first encryption ciphertext. Because the first encryption ciphertext is generated according to the first public key of the HSM and the second public key of the client, the first encryption ciphertext needs to be jointly decrypted by the HSM and the client, so as to obtain the key.

[0034] In this embodiment of this disclosure, the client generates an encryption key according to both the public key of the HSM and the public key of the client, so that the cloud server does not need to process the DEK in the KMS, and does not require the client to simultaneously store the data either, thereby reducing a computing load of the KMS and a data storage load of the cloud. For the HSM, because a layer-by-layer wrapping key structure is not used in this solution, computing operation costs for the HSM to generate, encrypt, and decrypt a hierarchical key are greatly lowered.

[0035] FIG. 5 is a schematic flowchart of another key management method according to an embodiment of this disclosure. This embodiment of this disclosure includes at least the following steps:

[0036] S501. A client selects a random number and a key according to an operation instruction inputted by a user. The random number and the key are both less than n, n being an order of a point on an elliptic curve.

[0037] In a specific implementation, for example, a user may access, through a client, a web page for creating a key on a cloud server. The client may receive an operation instruction inputted by the user through the web page for creating a key (for example, the user clicks or taps a button for generating a key on the web page), and execute code on the web page, to select any integer from a range from 1 to n-1 as a random number, and select an integer from a range from 1 to n-1 as a key, n being an order of a point on an elliptic curve. For example, for an elliptic curve $E : y^2 = x^3 + ax + b,$ if for a point P on the elliptic curve, a minimum positive integer n exists and makes n times P=O∞, n is referred to as an order of P; if n does not exist, P is of infinite order. O∞ is an infinity point on the elliptic curve.

[0038] Exemplarily, before the client selects a random number and a key, the HSM can perform initialization on the key management system. For example, when a user creates an account of the user on the cloud server, the HSM performs the initialization on the key management system. Specifically, the HSM may select, according to a selection instruction inputted by the user, one prime number q from a plurality of prime numbers, and then, select two non-negative integers *a* and *b* less than *q* from a plurality of integers, to make that $4a^3 + 27b^2 = 0 \bmod q$ is not established.

[0039] Therefore, all points (x, y) that satisfy the formula $E : y^2 = x^3 + ax + b$ and the infinite point O∞ form an elliptic curve, where x and *y* are integers ranging from 0 to p-1. Then, the HSM performs discretization on the elliptic curve, for example, $y^2 = x^3 + ax + b \bmod q$, to obtain all solutions $(x, y) \in Z_q$ and one infinite point O∞. Finally, the HSM selects one point P from all the solutions as any point on the elliptic curve, and an order of the point P is n. In addition, the HSM may receive the selection instruction inputted by the user and select one hash function from a plurality of hash functions, to make the hash function satisfy a condition $H : \{0,1\}^* \text{—} > \{0,1\}^{1024}$. The hash function may map any character string to a character string of a fixed length.

[0040] Then, the HSM may select an integer s_1 from [1, n-1] as a private key of the HSM, and then calculate a first public key of the HSM according to the private key of the HSM and a point P on an elliptic curve, where the first public key is $S_1 = s_1 gP$, and may be represented as (S_1,P).

[0041] During initialization of the key management system, the client may receive the point P on the elliptic curve and n (an order of P) sent by the HSM, then select, according to a selection instruction inputted by the user (where for example, the selection instruction is inputted through a web page provided by the cloud server or the HSM), select one integer s_2 from [1, n-1] as a private key of the client, and then calculate a second public key of the client according to the private key of the client and the point P on the elliptic curve, where the second public key $S_2 = s_2 gP$, and may be represented as (S_2,P). The user may download the private key of the client through the client, and store the private key in a memory of the client. The public key and the private key of the client are also referred to as a public key and a private key of the user.

[0042] S502. The client generates a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client;

[0043] In a specific implementation, the client may generate, by using the ECC, an encryption ciphertext *Ek* (that is, the first encryption ciphertext) of a key K according to both the first public key (S_1,P) of the HSM and the second public key (S_2,P) of the client, that is, $Ek = (r \cdot P,(S_1+S_2) \cdot r{+}k)$ (formula 1), where *r* is the random number, *k* is the key, S_1 is the first public key, and S_2 is the second public key. It can be learned from formula 1 that the encryption ciphertext *Ek* is determined jointly by the first public key of the HSM and the second public key of the client. $r \cdot P$ represents calculating a mapped point on the elliptic curve according to the random number *r* and the point *P* on the elliptic curve. The private key of the HSM is an integer selected by the HSM from a range from 1 to n-1, and the private key of the client is an integer selected by the client from a range from 1 to n-1, n being an order of the point on the elliptic curve.

[0044] S503. The client generates a symmetric key sequence according to the key and a preselected hash function, and encrypts cloud-stored data according to the symmetric key sequence to obtain a data ciphertext.

[0045] In a specific implementation, the client may call an encryption API, to generate, according to the key and a preselected hash function, a symmetric key sequence $(H(1\|k), H(2\|k),..., H(\mathrm{m}\|k))$ used for encrypting cloud-stored data, where *k* is the key, and *H* is the hash function, and encrypt, by using the ECC, cloud-stored data $(m1, m2,...m\mathrm{k})$ according to the symmetric key sequence $(H(1\|k), H(2\|k),..., H(\mathrm{m}\|k))$ to obtain a data ciphertext (Enc (m1), ..., Enc (mk)).

[0046] S504. The client transmits the first encryption ciphertext and the data ciphertext to a cloud server. The cloud server stores the first encryption ciphertext and the data ciphertext. For example, FIG. 4 is a schematic diagram of a storage format of a first encryption ciphertext and a data ciphertext according to an embodiment of this disclosure. The former portion is the first encryption ciphertext, and the latter portion is the data ciphertext. The data ciphertext can be decrypted by using the first encryption ciphertext. Because the first encryption ciphertext is generated according to the first public key of the HSM and the second public key of the client, the first encryption ciphertext needs to be jointly decrypted by the HSM and the client, so as to obtain the key.

[0047] S505. The client transmits a decryption request to the cloud server.

[0048] For example, when the user accesses cloud-stored data stored on the cloud server through a web page on the cloud server, the client transmits a decryption request to the cloud server. The decryption request, for example, may include a file name of the cloud-stored data, account information of the user, and the like.

[0049] S506. The cloud server transmits the first encryption ciphertext and the decryption request to the HSM.

[0050] The cloud server, for example, may obtain the first encryption ciphertext according to the account information of the user, and transmit the first encryption ciphertext and the decryption request together to the HSM.

[0051] S507. The HSM decrypts the first encryption ciphertext according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext.

[0052] In a specific implementation, after receiving the decryption request, the HSM decrypts the first encryption ciphertext by using the private key s_1 of the HSM, to calculate a second encryption ciphertext $\tilde{Ek}$, $\tilde{Ek}$, for example, being:

$$\tilde{Ek} = (r\cdot P, (S_1 + S_2)\cdot r + k - s_1\cdot r\cdot P) = (r\cdot P, S_2\cdot r + k), \qquad \text{(Formula 2)}$$

[0053] It can be learned from formula 2 that the second encryption ciphertext $\overline{\tilde{Ek}}$ is determined by only the second public key of the client.

[0054] Examplary, the HSM may receive a data ciphertext (Enc (m1), ..., Enc (mk)) transmitted by the cloud server, and transmit the data ciphertext (Enc (m1), ..., Enc (mk)) and the second encryption ciphertext $\tilde{Ek}$ to the client.

[0055] S508. The cloud server transmits a data ciphertext to the client. Exemplarily, S508 may be performed after S506 and before S507.

[0056] S509. The HSM transmits the second encryption ciphertext to the client.

[0057] S510. The client decrypts the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data.

[0058] In a specific implementation, the client may decrypt the second encryption ciphertext $r{\cdot}P, S_2\cdot r + k$ according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key $S_2\cdot r + k - s_2\cdot r\cdot P = k$, then call a decryption API, to generate the symmetric key sequence ($H(1\|k)$, $H(2\|k)$,..., $H(\mathrm{m}\|k)$) according to the key k and the hash function, where k is the key, and H is the hash function, and finally, decrypt the data ciphertext (Enc (m1), ..., Enc (mk)) according to the symmetric key sequence ($H(1\|k)$, $H(2\|k)$,..., $H(\mathrm{m}\|k)$) to obtain the cloud-stored data ($m1$, $m2$, ... mk).

[0059] FIG. 6 is a schematic diagram of storage load comparison according to an embodiment of this disclosure. In this embodiment of this disclosure, a length of a data ciphertext using the AES is 256 bits, and a length of an encryption ciphertext using the ECC is 384 bits. Since a layer-by-layer wrapping structure is used in a conventional solution, in a case of the same storage items, a storage load of this solution is less than a storage load of the conventional solution. Therefore, use of this embodiment of this disclosure can greatly improve storage efficiency of cloud-stored data.

[0060] In this embodiment of this disclosure, the HSM performs decryption according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext, and then, the client decrypts the second encryption ciphertext according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key. Therefore, an encryption ciphertext stored by the client on the cloud server can be encrypted only in a scenario in which the client and the HSM jointly provide a key, thereby guaranteeing the privacy and integrity of cloud-stored data.

[0061] FIG. 7 is a schematic structural diagram of a client according to an embodiment of this disclosure. In this embodiment of this disclosure, the client includes at least the following:

[0062] A selection module 701 is configured to select a random number and a key according to an operation instruction inputted by a user. The random number and the key are both less than n, n being an order of a point on an elliptic curve.

[0063] In a specific implementation, for example, a user may access, through a client, a web page for creating a key on a cloud server. The client may receive an operation instruction inputted by the user through the web page for creating a key (for example, the user clicks or taps a button for generating a key on the web page), and execute code on the web page, to select any integer from a range from 1 to n-1 as a random number, and select an integer from a range from 1 to n-1 as a key, n being an order of a point on an elliptic curve. For example, for an elliptic curve $E : y^2 = x^3 + ax + b$, if for a point P on the elliptic curve, a minimum positive integer n exists and makes n times P=O∞, n is referred to as an order of P, and if n does not exist, P is of infinite order. O∞ is an infinity point on the elliptic curve.

[0064] Exemplarily, before the client selects a random number and a key, the HSM can perform initialization on the key management system. For example, when a user creates an account of the user on the cloud server, the HSM performs the initialization on the key management system. Specifically, the HSM may select, according to a selection instruction inputted by the user, one prime number q from a plurality of prime numbers, and then, select two non-negative integers a and b less than q from a plurality of integers, to make that $4a^3 + 27b^2 = 0 \bmod q$ is not established. Therefore, all points (x, y) that satisfy the formula $E : y^2 = x^3 + ax + b$ and the infinite point O∞ form an elliptic curve, where x and y are integers ranging from 0 to p-1. Then, the HSM performs discretization on the elliptic curve, for example, $y^2 = x^3 + ax + b \bmod q$, to obtain all solutions $(x, y) \in Z_q$ and one infinite point O∞. Finally, the HSM selects one point P from all the solutions as any point on the elliptic curve, and an order of the point P is n. In addition, the HSM may receive the selection instruction inputted by the user and select one hash function from a plurality of hash functions, to make the hash function satisfy a condition $H : \{0,1\}^* \longrightarrow \{0,1\}^{1024}$. The hash function may map any character string to a character string of a fixed length.

[0065] Then, the HSM may select an integer s_1 from [1, n-1] as a private key of the HSM, and then calculate a first public key of the HSM according to the private key of the HSM and a point P on an elliptic curve, where the first public key is $S_1 = s_1 gP$, and is represented as (S_1, P).

[0066] During initialization of the key management system, the client may receive the point P on the elliptic curve and n (an order of P) sent by the HSM, then select, according to a selection instruction inputted by the user (for example, the selection instruction is inputted through a web page provided by the cloud server or the HSM), select one integer s_2 from [1, n-1] as a private key of the client, and then calculate a second public key of the client according to the private key of the client and the point P on the elliptic curve, where the second public key $S_2 = s_2 gP$, and is represented as (S_2, P). The user may download the private key of the client through the client, and store the private key in a memory of the client. The public key and the private key of the client are also referred to as a public key and a private key of the user.

[0067] The processing module 702 is configured to generate a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of a hardware security module (HSM), and the second public key being determined according to the point on the elliptic curve and a private key of the client.

[0068] In a specific implementation, an encryption ciphertext Ek (that is, the first encryption ciphertext) of a key K may be generated by using the ECC according to both the first public key (S_1, P) of the HSM and the second public key (S_2, P) of the client, that is, $Ek = (r \cdot P, (S_1 + S_2) \cdot r + k)$, where r is the random number, k is the key, S_1 is the first public key, and S_2 is the second public key. It can be learned from the formula that the encryption ciphertext Ek is determined jointly by the first public key of the HSM and the second public key of the client. $r \cdot P$ represents calculating a mapped point on the elliptic curve according to the random number r and the point P on the elliptic curve. The private key of the HSM is an integer selected by the HSM from a range from 1 to n-1, and the private key of the client is an integer selected by the client from a range from 1 to n-1, n being an order of the point on the elliptic curve.

[0069] A processing module 702 is further configured to generate a symmetric key sequence according to the key and a preselected hash function, and encrypt cloud-stored data according to the symmetric key sequence to obtain a data ciphertext.

[0070] In a specific implementation, the processing module 702 may call an encryption API, to generate, according to the key and a preselected hash function, a symmetric key sequence ($H(1\|k)$, $H(2\|k)$,..., $H(\mathrm{m}\|k)$) used for encrypting cloud-stored data, where k is the key, and H is the hash function, and encrypt, by using the ECC, cloud-stored data (m1, m2,... mk) according to the symmetric key sequence ($H(1\|k)$, $H(2\|k)$,..., $H(\mathrm{m}\|k)$) to obtain a data ciphertext (Enc (m1), ..., Enc (mk)).

[0071] A transmitting module 703 is configured to transmit the first encryption ciphertext and the data ciphertext to a cloud server. The cloud server stores the first encryption ciphertext and the data ciphertext. FIG. 4 is a schematic diagram of a storage format of a first encryption ciphertext and a data ciphertext according to an embodiment of this disclosure. The former portion is the first encryption ciphertext, and the latter portion is the data ciphertext. The data ciphertext can be decrypted by using the first encryption ciphertext. Because the first encryption ciphertext is generated according to the first public key of the HSM and the second public key of the client, the first encryption ciphertext needs to be jointly decrypted by the HSM and the client, so as to obtain the key.

[0072] Exemplarily, the transmitting module 703 is further configured to transmit a decryption request to the cloud server, the decryption request being used for instructing the cloud server to transmit the first encryption ciphertext to the HSM, the first encryption ciphertext being used for the HSM to decrypt the first encryption ciphertext according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext;

[0073] Exemplarily, the client further includes:

a receiving module 704, configured to receive the second encryption ciphertext transmitted by the HSM, and receive the data ciphertext transmitted by the cloud server. After receiving the decryption request, the HSM calculates a second encryption ciphertext $E\tilde{k}$ by using the private key of the HSM, $E\tilde{k} = (r \cdot P, (S_1 + S_2) \cdot r + k — s_1 \cdot r \cdot P) = (r \cdot P, S_2 \cdot r + k)$. It can be learned from the formula that the second encryption ciphertext $E\tilde{k}$ is determined by only the second public key of the client.

[0074] The processing module 702 is further configured to decrypt the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data.

[0075] Further, the processing module 702 is further configured to decrypt the second encryption ciphertext according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key; generate the symmetric key sequence according to the key and the hash function; and decrypt the data ciphertext according to the symmetric key sequence to obtain the cloud-stored data.

[0076] In a specific implementation, the client decrypts the second encryption ciphertext $r \cdot P$, $S_2 \cdot r + k$ according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key $S_2 \cdot r + k — s_2 \cdot r \cdot P = k$, then calls a decryption API, to generate the symmetric key sequence ($H(1\|k)$, $H(2\|k)$,..., $H(\mathrm{m}\|k)$) according to the key k and the hash function, where k is the key, and H is the hash function, and finally, decrypts the data ciphertext (Enc (m1), ..., Enc (mk)) according to the symmetric key sequence to obtain the cloud-stored data *(m1,m2,...m*k).

[0077] Further, for a specific implementation of the client in this embodiment of this disclosure, refer to operation steps of the client in the foregoing method embodiments.

[0078] FIG. 8 is a schematic structural diagram of a cloud server according to an embodiment of this disclosure. The cloud server in this embodiment of this disclosure includes at least the following:

[0079] A receiving module 801 is configured to receive a first encryption ciphertext and a data ciphertext transmitted by a client, the first encryption ciphertext being generated according to a random number, a key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, the second public key being determined according to the point on the elliptic curve and a private key of the client, the data ciphertext being obtained by encrypting cloud-stored data according to a symmetric key sequence, and the symmetric key sequence being generated according to the key and a preselected hash function.

[0080] Exemplarily, the receiving module 801 is configured to receive a decryption request transmitted by the client.

[0081] A transmitting module 802 is configured to transmit the first encryption ciphertext to the HSM, the first encryption ciphertext being used for instructing the HSM to decrypt the first encryption ciphertext according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext.

[0082] The transmitting module 802 is configured to transmit the data ciphertext to the client, the data ciphertext being encrypted by the client according to the second encryption ciphertext to obtain the cloud-stored data.

[0083] Further, for a specific implementation of the cloud server in this embodiment of this disclosure, refer to operation steps of the cloud server in the foregoing method embodiments.

[0084] FIG. 9 is a schematic structural diagram of another client according to an embodiment of this disclosure. As shown in FIG. 9, the client may include: at least one processor 901, at least one communication interface 902, at least one memory 903, and at least one communication bus 904.

[0085] The processor 901 may be a central processing unit, a general-purpose processor, a digital signal processor, an application-specific integrated circuit, a field programmable gate array or another programmable logic device, a transistor logic device, a hardware component, or any combination thereof. The processor may implement or perform various examples of logic blocks, modules, and circuits described with reference to content disclosed in this disclosure. The processor may alternatively be a combination to implement a computing function, for example, may be a combination of one or more microprocessors, a combination of a digital signal processor and a microprocessor, or the like. The communication bus 904 may be a peripheral component interconnect (PCI) bus, an extended industry standard architecture (EISA) bus, or the like. The bus may be classified as an address bus, a data bus, a control bus, or the like. For ease of description, the bus in FIG. 9 is represented by using only one bold line, but this does not indicate that there is only one bus or one type of bus. The communication bus 904 is configured to implement connection and communication between the components. The communication bus 902 of the device in this embodiment of this disclosure is configured to communicate signaling or data with another node device. The memory 903 may include a volatile memory, for example, a non-volatile dynamic random access memory (NVRAM), a phase change random access memory (PRAM), or a magnetoresistive random access memory (MRAM), and may further include a non-volatile memory, for example, at least one magnetic disk storage device, an electrically erasable programmable read-only memory (EEPROM), a flash memory device such as a NOR flash memory or a NAND flash memory, or a semiconductor device such as a solid state disk (SSD). Exemplarily, the memory 903 may alternatively be at least one storage apparatus far away from the processor 901. Exemplarily, the memory 903 may further store a set of program code. Exemplarily, the processor 901 may further execute a program stored in the memory 903, including the following steps:

selecting a random number and a key according to an operation instruction inputted by a user;

generating a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client;

generating a symmetric key sequence according to the key and a preselected hash function, and encrypting cloud-stored data according to the symmetric key sequence to obtain a data ciphertext; and

transmitting the first encryption ciphertext and the data ciphertext to a cloud server.

[0086] Exemplarily, the processor 901 may further be configured to perform the following steps:

transmitting a decryption request to the cloud server, the decryption request being used for instructing the cloud server to transmit the first encryption ciphertext to the HSM, the first encryption ciphertext being used for instructing the HSM to decrypt the first encryption ciphertext according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext;

receiving the second encryption ciphertext transmitted by the HSM, and receiving the data ciphertext transmitted by the cloud server; and

decrypting the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data.

[0087] Exemplarily, the processor 901 may further be configured to perform the following steps:

decrypting the second encryption ciphertext according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key;

generating the symmetric key sequence according to the key and the hash function; and

decrypting the data ciphertext according to the symmetric key sequence to obtain the cloud-stored data.

[0088] The random number and the key are both less than n, n being an order of the point on the elliptic curve.

[0089] The private key of the HSM is an integer selected by the HSM from a range from 1 to n-1, and the private key of the client is an integer selected by the client from a range from 1 to n-1, n being an order of the point on the elliptic curve.

[0090] Further, the processor may further cooperate with the memory and the communication interface to perform operations of the client in the foregoing embodiments of this disclosure.

[0091] FIG. 10 is a schematic structural diagram of another cloud server according to an embodiment of this disclosure. As shown in the figure, the cloud server may include: at least one processor 1001, at least one communication interface 1002, at least one memory 1003, and at least one communication bus 1004.

[0092] The processor 1001 may be various types of processors mentioned above. The communication bus 1004 may be a peripheral component interconnect (PCI) bus, an extended industry standard architecture (EISA) bus, or the like. The bus may be classified as an address bus, a data bus, a control bus, or the like. For ease of description, the bus in FIG. 10 is represented by using only one bold line, but this does not indicate that there is only one bus or one type of bus. The communication bus 1004 is configured to implement connection and communication between the components. The communication bus 1002 of the device in this embodiment of this disclosure is configured to communicate signaling or data with another node device. The memory 1003 may be various types of memories mentioned above. Exemplarily, the memory 1003 may alternatively be at least one storage apparatus far away from the processor 1001. The memory 1003 stores a set of program code. Exemplarily, the processor 1001 executes a program stored in the memory 1003, including the following steps:

receiving a first encryption ciphertext and a data ciphertext transmitted by a client, the first encryption ciphertext being generated according to a random number, a key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of an HSM, the second public key being determined according to the point on the elliptic curve and a private key of the client, the data ciphertext being obtained by encrypting cloud-stored data according to a symmetric key sequence, and the symmetric key sequence being generated according to the key and a preselected hash function; and

storing the first encryption ciphertext and the data ciphertext.

[0093] Exemplarily, the processor 1001 may further be configured to perform the following steps:

receiving a decryption request transmitted by the client;

transmitting the first encryption ciphertext to the HSM, the first encryption ciphertext being decrypted by the HSM according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext; and

transmitting the data ciphertext to the client, the data ciphertext being encrypted by the client according to the second encryption ciphertext to obtain the cloud-stored data.

[0094] Further, the processor may further cooperate with the memory and the communication interface to perform operations of the cloud server in the foregoing embodiments of this disclosure.

[0095] All or some of the foregoing embodiments may be implemented by using software, hardware, firmware, or any combination thereof. When software is used for implementation, all or some of the embodiments may be implemented in a form of a computer program product. The computer program product includes one or more computer instructions. When the computer program instructions are loaded and executed on a computer, all or some of the procedures or functions according to the embodiments of this disclosure are generated. The computer may be a general-purpose computer, a dedicated computer, a computer network, or other programmable apparatuses. The computer instructions may be stored in a computer-readable storage medium or may be transmitted from a computer-readable storage medium to another computer-readable storage medium. For example, the computer instructions may be transmitted from a website, computer, server, or data center to another website, computer, server, or data center in a wired (for example, a coaxial cable, an optical fiber, or a digital subscriber line (DSL)) or wireless (for example, infrared, radio, or microwave) manner. The computer-readable storage medium may be any usable medium accessible by a computer, or a data storage device, such as a server or a data center, integrating one or more usable media. The usable medium may be a magnetic medium (for example, a soft disk, a hard disk, or a magnetic tape), an optical medium (for example, a DVD), a semiconductor medium (for example, a solid state disk (SSD)), or the like.

[0096] The foregoing specific implementations further describe the objectives, technical solutions, and beneficial effects of this disclosure in detail. Any modification, equivalent replacement, or improvement made without departing from the spirit and principle of this disclosure shall fall within the protection scope of this disclosure.

Claims

1. A key management method, **characterized by** comprising:

selecting, by a client, a random number and a key according to an operation instruction inputted by a user;
generating, by the client, a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of a hardware security module, HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client;
generating, by the client, a symmetric key sequence according to the key and a preselected hash function, and encrypting cloud-stored data according to the symmetric key sequence to obtain a data ciphertext; and
transmitting, by the client, the first encryption ciphertext and the data ciphertext to a cloud server.

2. The method according to claim 1, **characterized in that** after the transmitting, by the client, the first encryption ciphertext and the data ciphertext to a cloud server, the method further comprises:

transmitting, by the client, a decryption request to the cloud server, the decryption request being used for instructing the cloud server to transmit the first encryption ciphertext to the HSM, the first encryption ciphertext being decrypted by the HSM according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext;
receiving, by the client, the second encryption ciphertext transmitted by the HSM, and the data ciphertext transmitted by the cloud server; and
decrypting, by the client, the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data.

3. The method according to claim 2, **characterized in that** the decrypting, by the client, the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data comprises:

decrypting, by the client, the second encryption ciphertext according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key;
generating, by the client, the symmetric key sequence according to the key and the hash function; and
decrypting, by the client, the data ciphertext according to the symmetric key sequence to obtain the cloud-stored data.

4. The method according to any one of claims 1 to 3, **characterized in that** the random number and the key are both less than n, n being an order of the point on the elliptic curve.

5. The method according to any one of claims 1 to 3, **characterized in that** the private key of the HSM is an integer selected by the HSM from a range from 1 to n-1, and the private key of the client is an integer selected by the client from a range from 1 to n-1, n being an order of the point on the elliptic curve.

6. A key management method, **characterized by** comprising:

receiving, by a cloud server, a first encryption ciphertext and a data ciphertext transmitted by a client, the first encryption ciphertext being generated according to a random number, a key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of a hardware security module, HSM, the second public key being determined according to the point on the elliptic curve and a private key of the client, the data ciphertext being obtained by encrypting cloud-stored data according to a symmetric key sequence, and the symmetric key sequence being generated according to the key and a preselected hash function; and
storing, by the cloud server, the first encryption ciphertext and the data ciphertext.

7. The method according to claim 6, **characterized in that** after the receiving, by a cloud server, a first encryption ciphertext and a data ciphertext transmitted by a client, the method further comprises:

receiving, by the cloud server, a decryption request transmitted by the client;
transmitting, by the cloud server, the first encryption ciphertext to the HSM, the first encryption ciphertext being decrypted by the HSM according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext; and
transmitting, by the cloud server, the data ciphertext to the client, the data ciphertext being encrypted by the client according to the second encryption ciphertext to obtain the cloud-stored data.

8. A client, **characterized by** comprising:

a selection module, configured to select a random number and a key according to an operation instruction inputted by a user;
a processing module, configured to generate a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key, the first public key being determined according to a point on an elliptic curve and a private key of a hardware security module ,HSM, and the second public key being determined according to the point on the elliptic curve and a private key of the client,
the processing module being further configured to generate a symmetric key sequence according to the key and a preselected hash function, and encrypt cloud-stored data according to the symmetric key sequence to obtain a data ciphertext; and
a transmitting module, configured to transmit the first encryption ciphertext and the data ciphertext to a cloud server.

9. The client according to claim 8, **characterized in that**

the transmitting module is further configured to transmit a decryption request to the cloud server, the decryption request being used for instructing the cloud server to transmit the first encryption ciphertext to the HSM, the first encryption ciphertext being decrypted by the HSM according to the private key of the HSM, the point on the elliptic curve, and the random number to obtain a second encryption ciphertext;
the client further comprises:

a receiving module, configured to receive the second encryption ciphertext transmitted by the HSM, and the data ciphertext transmitted by the cloud server; and
the processing module is further configured to decrypt the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data.

10. The client according to claim 9, **characterized in that**
the processing module is further configured to decrypt the second encryption ciphertext according to the private key of the client, the random number, and the point on the elliptic curve to obtain the key; generate the symmetric key sequence according to the key and the hash function; and decrypt the data ciphertext according to the symmetric key sequence to obtain the cloud-stored data.

11. A computer-readable storage medium, **characterized by** storing a computer program, the computer program comprising program instructions, the program instructions, when executed by a processor, causing the method according to any one of claims 1, 6, and 7 to be performed.

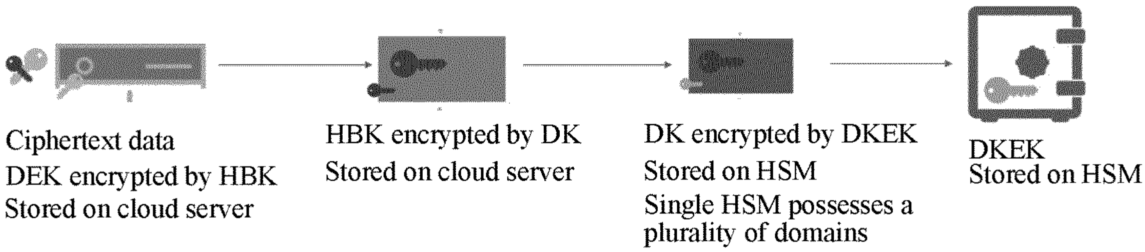
Ciphertext data
DEK encrypted by HBK
Stored on cloud server
HBK encrypted by DK
Stored on cloud server
DK encrypted by DKEK
Stored on HSM
Single HSM possesses a plurality of domains
DKEK
Stored on HSM

FIG. 1

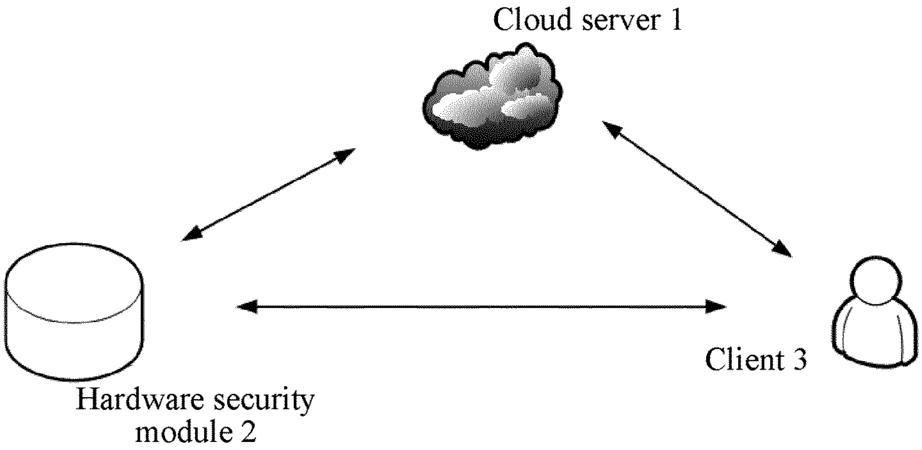
Cloud server 1
Hardware security module 2
Client 3

FIG. 2

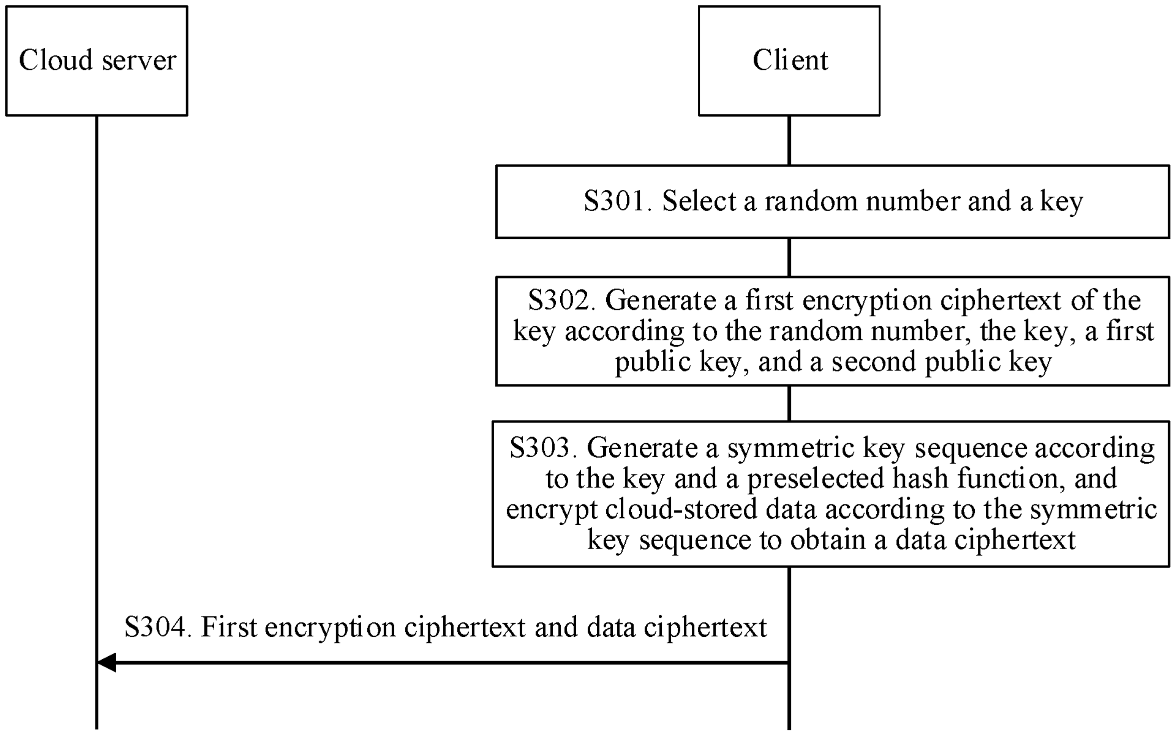
Cloud server
Client
S301. Select a random number and a key
S302. Generate a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key
S303. Generate a symmetric key sequence according to the key and a preselected hash function, and encrypt cloud-stored data according to the symmetric key sequence to obtain a data ciphertext
S304. First encryption ciphertext and data ciphertext

FIG. 3

Encryption ciphertext *Ek*	Data ciphertext (Enc(m1), ..., Enc(mk))

FIG. 4

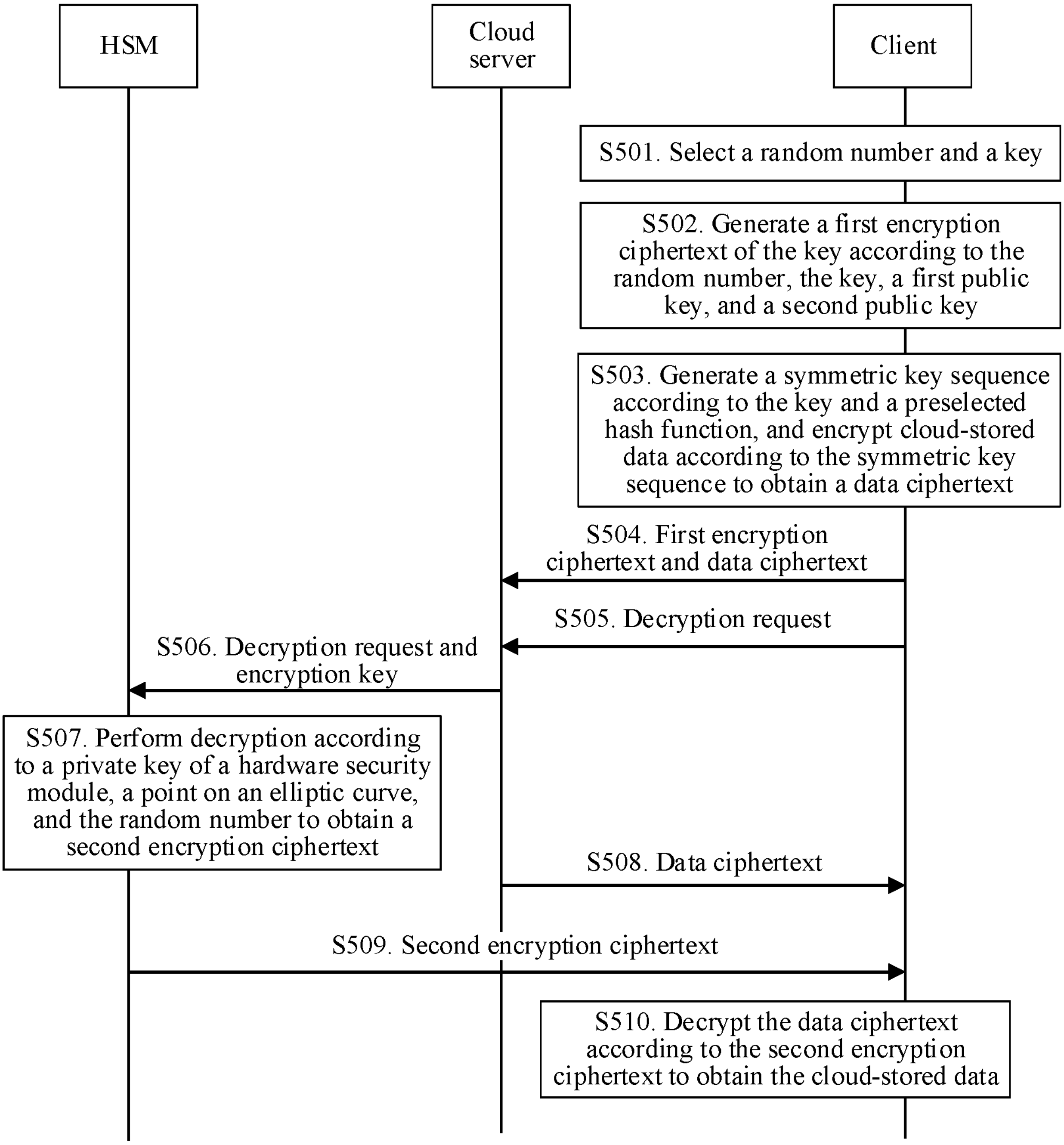
HSM
Cloud server
Client
S501. Select a random number and a key
S502. Generate a first encryption ciphertext of the key according to the random number, the key, a first public key, and a second public key
S503. Generate a symmetric key sequence according to the key and a preselected hash function, and encrypt cloud-stored data according to the symmetric key sequence to obtain a data ciphertext
S504. First encryption ciphertext and data ciphertext
S505. Decryption request
S506. Decryption request and encryption key
S507. Perform decryption according to a private key of a hardware security module, a point on an elliptic curve, and the random number to obtain a second encryption ciphertext
S508. Data ciphertext
S509. Second encryption ciphertext
S510. Decrypt the data ciphertext according to the second encryption ciphertext to obtain the cloud-stored data

FIG. 5

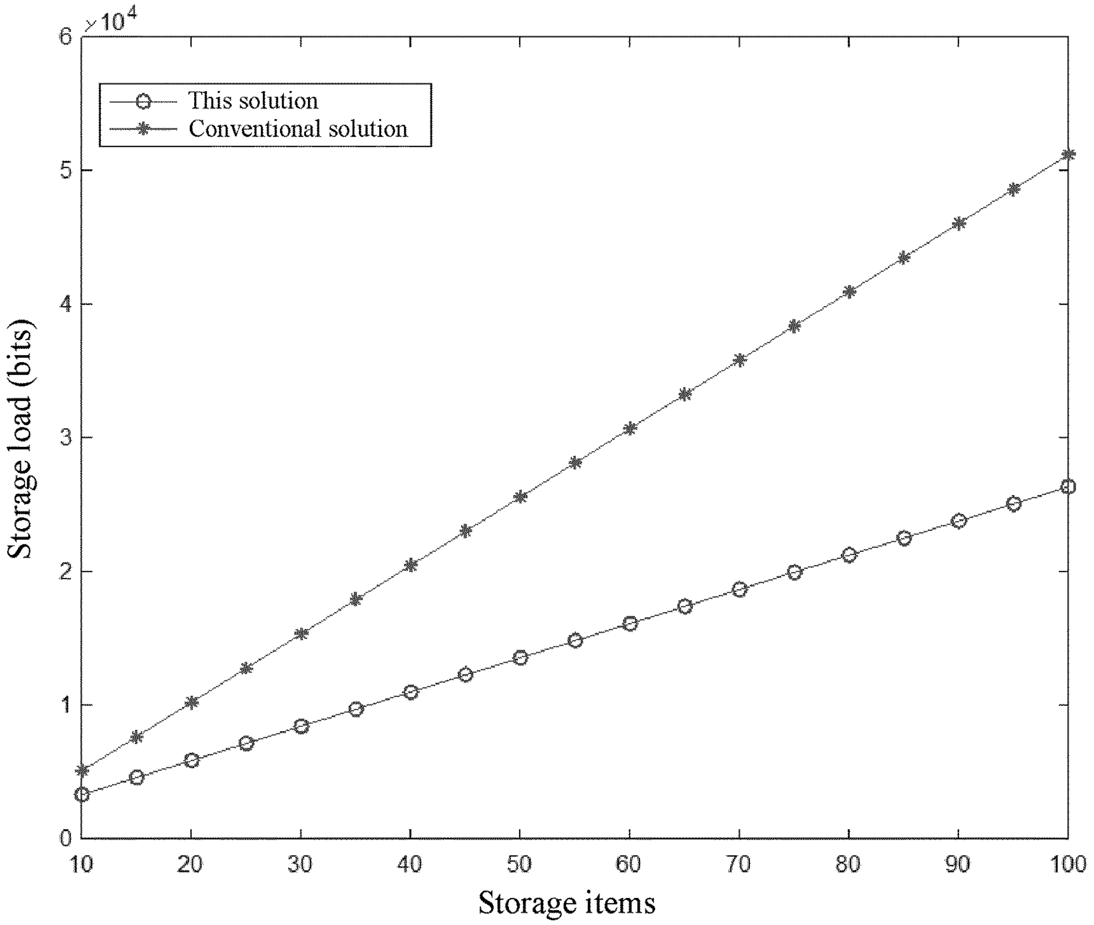

FIG. 6

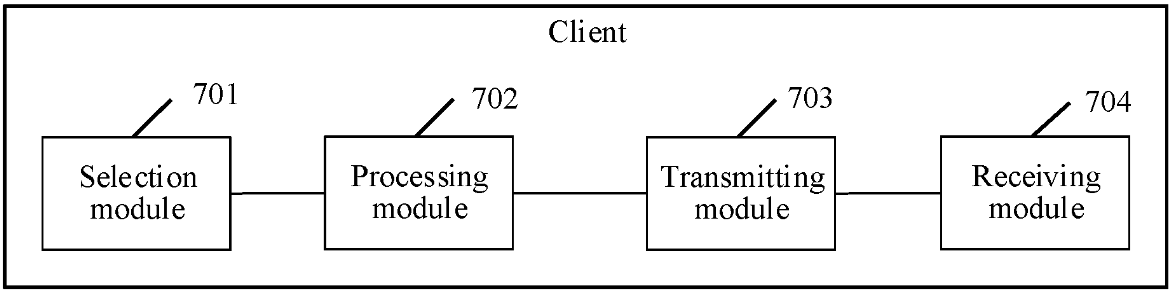

FIG. 7

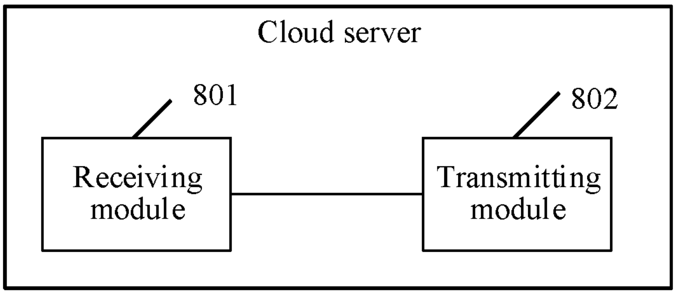

FIG. 8

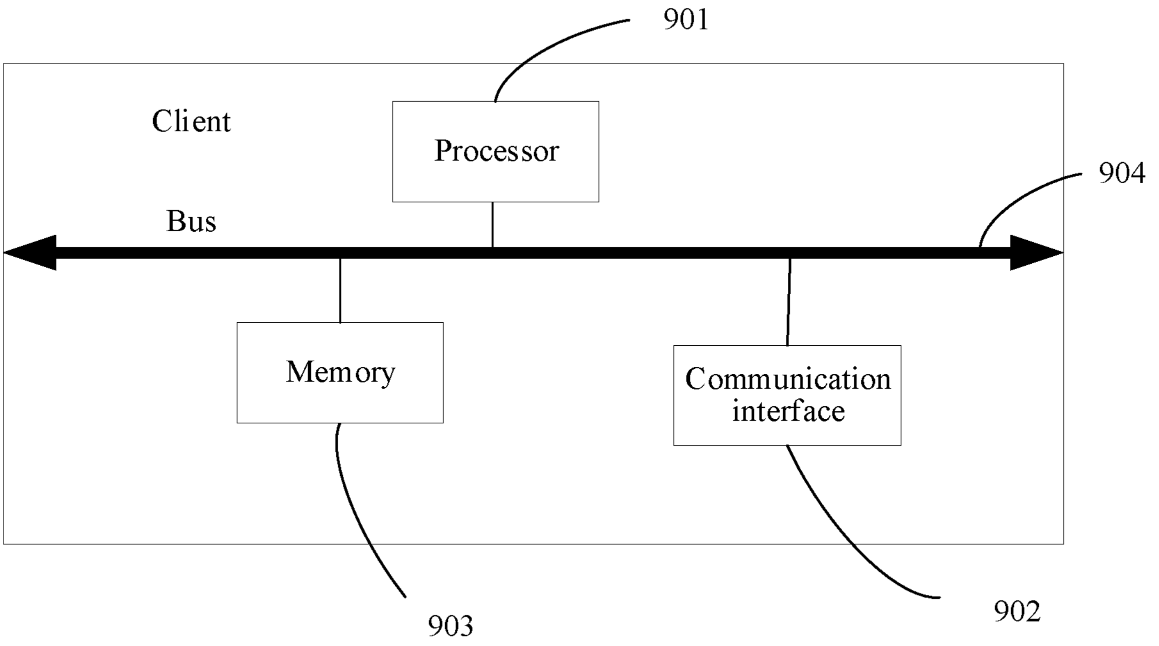
901
Client
Processor
904
Bus
Memory
Communication interface
903
902

FIG. 9

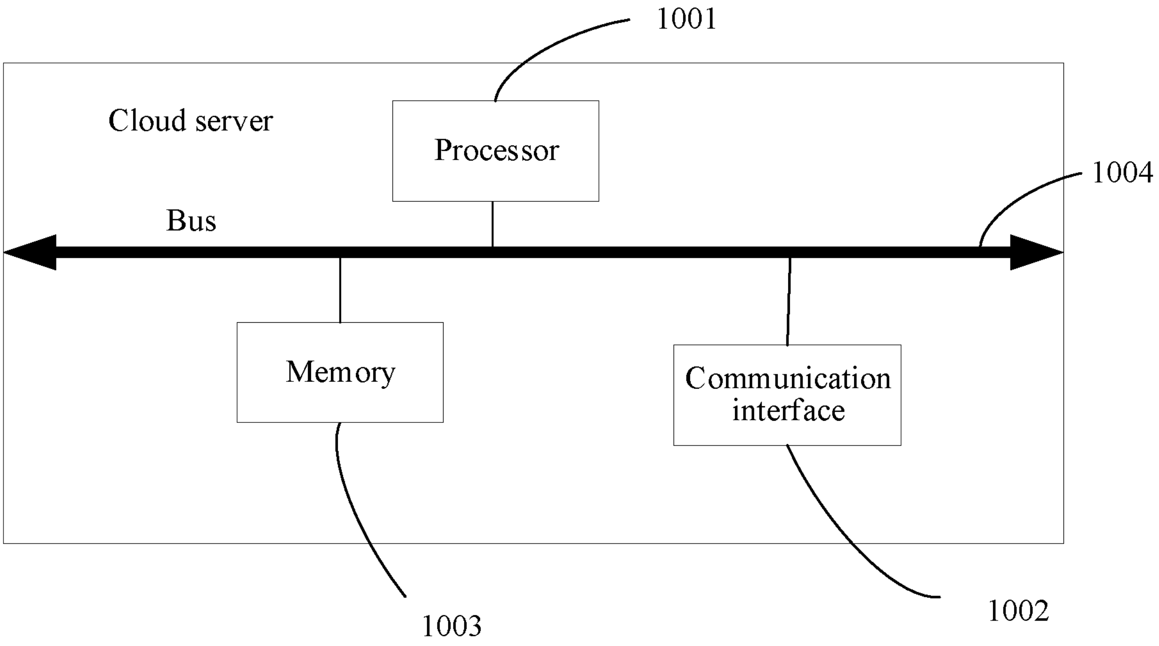
1001
Cloud server
Processor
1004
Bus
Memory
Communication interface
1003
1002

FIG. 10

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2020/091002

A. CLASSIFICATION OF SUBJECT MATTER

H04L 9/00(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

VEN, CNKI, CNABS, CNTXT, USTXT, WOTXT, EPTXT, DWPI: 硬件安全模块, 随机数, 私钥, 客户端, hsm, client, random number, private key

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 110417726 A (TENCENT TECHNOLOGY SHENZHEN CO., LTD.) 05 November 2019 (2019-11-05) description, pages 4-13	1-11
A	US 2018212762 A1 (SALESFORCE COM INC.) 26 July 2018 (2018-07-26) entire document	1-11
A	CN 105119894 A (SHANGHAI HUIYIN INFORMATION TECHNOLOGY CO., LTD.) 02 December 2015 (2015-12-02) entire document	1-11

□ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search **03 August 2020**	Date of mailing of the international search report **14 August 2020**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China**	Authorized officer
Facsimile No. **(86-10)62019451**	Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2020/091002

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 110417726 A	05 November 2019	None	
US 2018212762 A1	26 July 2018	US 10637658 B2	28 April 2020
CN 105119894 A	02 December 2015	CN 105119894 B	25 May 2018

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- CN 201910445155 **[0001]**